

**计算机网络**

**课程实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 实验名称 | HTTP 代理服务器的设计与实现 | | | | | |
| 姓名 | 汤添凝 | | 院系 | 计算机科学有技术学院 | | |
| 班级 | 1703201 | | 学号 | 1170300728 | | |
| 任课教师 | 李全龙 | | 指导教师 | 李全龙 | | |
| 实验地点 | 格物214 | | 实验时间 | 10/26 | | |
| 实验课表现 | 出勤、表现得分(10) |  | 实验报告  得分(40) |  | 实验总分 |  |
| 操作结果得分(50) |  |
| 教师评语 | | | | | | |
|  | | | | | | |

****

|  |
| --- |
| 实验目的： |
| 熟悉并掌握 Socket 网络编程的过程与技术；深入理解 HTTP 协议，  掌握 HTTP 代理服务器的基本工作原理；掌握 HTTP 代理服务器设计与  编程实现的基本技能。 |
| 实验内容： |
| (1) 设计并实现一个基本 HTTP 代理服务器。要求在指定端口（例如  **8080）**接收来自客户的 HTTP 请求并且根据其中的 URL 地址访问该地址  所指向的 HTTP 服务器（原服务器），接收 HTTP 服务器的响应报文，并  将响应报文转发给对应的客户进行浏览。  (2) 设计并实现一个支持 Cache 功能的 HTTP 代理服务器。要求能缓  存原服务器响应的对象，并能够通过修改请求报文（添加 if-modified-since  头行），向原服务器确认缓存对象是否是最新版本。（选作内容，加分项  目，可以当堂完成或课下完成）  (3) 扩展 HTTP 代理服务器，支持如下功能：（选作内容，加分项目，  可以当堂完成或课下完成）  a) 网站过滤：允许/不允许访问某些网站；  b) 用户过滤：支持/不支持某些用户访问外部网站；  c) 网站引导：将用户对某个网站的访问引导至一个模拟网站（钓  鱼）。 |
| 实验过程： |
| 1. **实验所需知识总结**   本次实验要求设计并实现一个基本HTTP代理服务器。首先我们需要了解一下HTTP协议的细节才有可能完成   * 1. HTTP协议概述   首先，HTTP协议是应用层协议，全称为HyperText TransferProtocol，即超文本传输协议。同其他应用层协议一样，是为了实现某一类具体应用的协议，并由某一运行在用户空间的应用程序来实现其功能。HTTP是一种协议规范，这种规范记录在文档上，实际应用中由程序真正实现HTTP协议进行通信。  HTTP协议是基于C/S架构进行通信的。其中HTTP协议的客户端实现程序需要完成请求、接受、展示Web对象等功能；而服务器端则需要响应客户的请求，并发送对应的对象。Web服务是基于TCP的，因此为了能够随时响应客户端的请求，Web服务器需要监听80/TCP端口。  流程上而言，首先服务器在80端口等待客户的请求，随后客户浏览器发起到服务器的TCP连接（同时创建套接字Socket），当服务器接受来自浏览器的TCP连接，服务器利用accept报文同意建立链接，之后浏览器（HTTP客户端）与Web服务器（HTTP服务器）交换HTTP消息，当所有需求的报文接收和发送完毕，双方确认后关闭TCP连接。  下面是利用socket API设计的HTTP服务中客户端和服务器程序基本调用流程图：     * 1. HTTP协议的数据报构成   大致了解了HTTP之后我们依然不能设计出我们的HTTP代理服务器，因为我们在HTTP代理服务器设计中需要大量地操作HTTP的报文数据，这需要我们对HTTP的消息报文结构有所了解。  这里给出HTTP请求消息的报文格式：  IMG_256  以及HTTP服务器端相应消息的报文样例：  IMG_256   * 1. HTTP代理服务器设计   接下来我们需要考虑一下HTTP代理服务器究竟如何设计实现。  代理服务器，俗称“翻墙软件”，允许一个网络终端（一般为客户端） 通过这个服务与另一个网络终端（一般为服务器）进行非直接的连接。如图所示，为普通 Web 应用通信方式与采用代理服务器的通信方式的对比。  ds  代理服务器在指定端口（例如 8080）监听浏览器的访问请求（需要在客户端浏览器进行相应的设置），接收到浏览器对远程网站的浏览请求时，代理服务器开始在代理服务器的缓存中检索 URL 对应的对象（网页、图像等对象），找到对象文件后，提取该对象文件的最新被修改时间；代理服务器程序在客户的请求报文首部插入<If-Modified-Since: 对象文件的最新被修改时间>，并向原 Web 服务器转发修改后的请求报文。如果代理服务器没有该对象的缓存，则会直接向原服务器转发请求报文，并将原服务器返回的响应直接转发给客户端，同时将对象缓存到代理服务器中。代理服务器程序会根据缓存的时间、大小和提取记录等对缓存进行清理。   * 1. Cache功能   Cache其实就是我们通常所说的缓存技术，但是不同情境下的缓存要求是不一样的，这里探讨一下HTTP代理服务器的缓存应该如何设计实现。  首先回忆一下过去所学的CPU中的缓存。缓存的工作原理是当CPU要读取一个数据时，首先从CPU缓存中查找，找到就立即读取并送给CPU处理；没有找到，就从速率相对较慢的内存中读取并送给CPU处理，同时把这个数据所在的数据块调入缓存中，可以使得以后对整块数据的读取都从缓存中进行，不必再调用内存。  这里HTTP代理服务器的缓存原理和思路是类似的。首先我们将过去的服务器相应消息保存下来，当客户请求一个数据时首先查找缓存中是否已经包括了这个请求的响应消息cache，如果没有，直接向服务器发送该请求，若有，从缓存中找到Date信息，并在请求报文中加入“If\_Modified\_Since”语句，询问服务器这个请求的url在date时间之后有没有被修改过，如果没有被修改过，服务器会发送“304”，这时候代理服务器直接将缓存中的响应发送给客户即可；如果被修改了，那么服务器会照常发送“200”，代理服务器转发消息，并进行缓存的更新。  IMG_256   1. 实验内容完成   2.1 关键技术：基本代理的实现  解决方案：主要参考以下几个函数：    2.1.1 BOOL InitSocket()  函数功能为创建一个新的套接字。套接字是TCP协议依靠的主要API。首先加载套接字库，使用以下几个socket函数socket(AF\_INET, SOCK\_STREAM, 0);  bind(ProxyServer, (SOCKADDR\*)&ProxyServerAddr, sizeof(SOCKADDR))；和  listen(ProxyServer, SOMAXCONN)实现了服务器流程中的socket和bind和listen，从而创建了一个绑定好的套接字，以备使用。  2.1.2 unsigned int \_\_stdcall ProxyThread(LPVOID lpParameter)  这个函数是代码核心的线程执行函数，它实现了从客户端接收请求报文，向服务器发送请求报文，从服务器接收响应报文，向客户端送响应报文整个代理服务器单个线程需要完成的任务。  具体而言，线程首先通过ParseHttpHead函数对请求报文头部进行解析，得到头部信息；然后将得到信息交付给ConnectToServer函数，从而与目标服务器建立链接；连接成功后，线程将请求（修改或未修改过的）报文发送过去，并等待接收响应报文；最后，线程发送响应报文给浏览器即可。  同时，这一函数中还需要考虑实现代理服务器的其他关键技术，比如对网站的过滤，钓鱼网站以及缓存检查等。  2.1.3 BOOL ParseHttpHead(char \*buffer, HttpHeader \* httpHeader)  这一函数的作用是对TCP请求报文的头部文件进行解析，尝试获取请求报文中的method, url, host和cookie等信息，以便后续利用ConnectToServer函数使代理服务器与目标服务器建立连接。  2.1.4 BOOL ConnectToServer(SOCKET \*serverSocket, char \*host)  该函数的功能是根据主机，与目标服务器使用socket创建套接字，诉后connect连接至目标服务器，等待服务器的响应消息。  2.2 关键技术：代理服务器设置cahce实现方式  解决方案：如前文所言，代理服务器在线程执行函数中完成这一任务：代理服务器第一次和客户端通信时会保留Cache；当客户端再次请求本地存在的 cache页面时，代理服务器会通过MakeDateIfNeed函数提取出相关的Date信息，并利用 If-Modified-Since将Date信息加入请求报文，再将修改过的报文发送给目标服务器，让目标服务器端进行验证。目标服务器通过这个时间戳判断客户端的页面是否是最新的，如果不是最新的，则返回“200”以及新的内容；而如果是最新的，则返回“304”告诉代理服务器其本地 cache 的页面已经是最新的了，这时候代理服务器将本地Cache直接发送给客户端即可。  2.3 关键技术：网站过滤  解决方案：在线程执行函数中将解析出的HTTP头部中的host与过滤表进行比对，如果发现host在过滤列表中，表示访问的网站被禁止访问，此时goto error。  IMG_256  2.4 关键技术：用户过滤  解决方案： 由于用户是在所有线程开始之前就需要判断的，所以主函数中，当建立起客户端和代理服务器的连接时，得到客户端的 ip地址，就需要与与被过滤用户ip比较，如果相同，则跳过建立。  IMG_256  2.5 关键技术：钓鱼网站  解决方案：类似于网站过滤，在线程执行函数中将请求报文头部中的host与被引导网站进行比较，如果相同则将报文中的所有host相关内容改为钓鱼网站的host，再将修改后的请求报文发送给目标服务器。  IMG_256 |
| 实验结果： |
| 第一步：设置浏览器的代理服务器，这里地址设置为：127.0.0.1，端口设置为：10240。  QQ图片20191030220914  第二步：启动代理服务器程序，尝试在浏览器上访问哈工大官网www.hit.edu.cn，程序部分显示如下  QQ图片20191030221021  浏览器成功打开哈工大官网  QQ图片20191030221038  第三步：测试网页过滤，过滤列表中有www.7k7k.com，发现无法打开网页  QQ图片20191030221117  程序显示如下，关注红圈部分：  QQ图片20191030221131  第四步：测试钓鱼网站，钓鱼网站图中包含www.4399.com转到today.hit.edu.cn的钓鱼关系，尝试在浏览器中访问4399网页，发现浏览器中打开了今日哈工大的页面，  QQ图片20191030221217  程序控制台中输出如下：  QQ图片20191030222122  第五步：测试cache，从之前的截图中可以看到各种“接受到304”，“已缓存”等消息，同时文件列表中也存入了各个网站的cache信息  QQ图片20191030221240  QQ图片20191030222536  第六步：测试用户过滤，开启过滤之后，用户发送的请求不予处理  QQ图片20191030222427 |
| 心得体会： |
| 本次实验内容充实紧凑，且有一定难度与复杂度，但在老师、助教、同学以及个人自学的多重努力下成功完成了实验。  本次实验为“HTTP 代理服务器的设计与实现  ”，以设计代理服务器的形式逐步教会我们HTTP协议的含义细节，协议信息构造，缓存设计实现，效果出众。在此前，我从来没有考虑过诸如“数据报文到底是什么格式的？”、“缓存是存在文件好还是存在数据结构中好？”这类问题，经过了这次实验后，我对这些HTTP协议设计细节有了更深入的了解，在逐渐完成实验任务的同时，开拓了计算思维，以至于最后的服务器调试在运行了程序后很快便意识到了问题所在。我想这大概就是身为“计算机科学有技术学院”的学生相比其他专业学生对于计算机认识更深入的地方之一了。 |
|  |
|  |